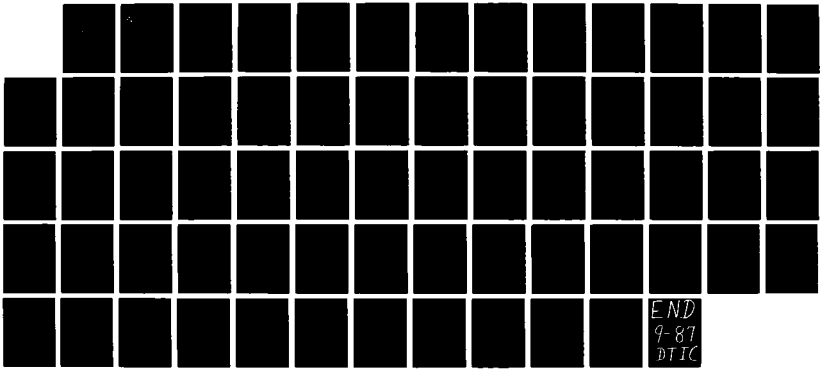
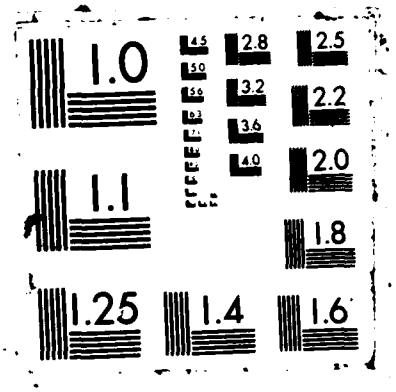
AD-A183 476 ERROR CONTROL OPTION FOR GROUP 3 FACSIMILE EQUIPMENT (U) DELTA INFORMATION SYSTEMS INC HORSHAM PA FEB 87 NCS-TIB-87-4 DCA100-83-C-0047 1/1

UNCLASSIFIED F/G 25/5 NL




1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6

NCS TIB 87-4

NATIONAL COMMUNICATIONS SYSTEM

TECHNICAL INFORMATION BULLETIN 87-4

ERROR CONTROL OPTION FOR GROUP 3 FACSIMILE EQUIPMENT

FEBRUARY 1987

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE

ADA183476

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION / AVAILABILITY OF REPORT Distribution Unlimited
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S) NCS TIB 87-4	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Delta Information Systems, Inc. 6b. OFFICE SYMBOL (*If applicable*)	7a. NAME OF MONITORING ORGANIZATION
6c. ADDRESS (*City, State, and ZIP Code*) Horsham Business Center, Bldg. 3 300 Welsh Road Horsham, PA 19044	7b. ADDRESS (*City, State, and ZIP Code*)
8a. NAME OF FUNDING / SPONSORING ORGANIZATION National Communications System 8b. OFFICE SYMBOL (*If applicable*) NCS-TS	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER DCA100-83-C-0047
8c. ADDRESS (*City, State, and ZIP Code*) Office of Technology & Standards Washington, D.C. 20305-2010	10. SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO. P00010/8 / WORK UNIT ACCESSION NO.

11. TITLE (*Include Security Classification*)
Error Control Option for Group 3 Facsimile Equipment

12. PERSONAL AUTHOR(S)

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (*Year, Month, Day*)	15. PAGE COUNT
Final	FROM _____ TO _____	February 1987	62

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (*Continue on reverse if necessary and identify by block number*)
FIELD	GROUP	SUB-GROUP	Group 3 Facsimile; Error Control; Public Switched Telephone Network (PSTN)

19. ABSTRACT (*Continue on reverse if necessary and identify by block number*)

Group 3 facsimile equipment, which meets CCITT Recommendations T.4 and T.30 (Federal Standards 1062 and 1063, respectively), is now used extensively throughout the world and throughout the U.S. Government. This equipment has no error control capability; the output image quality and transmission bit rate are totally dependent upon the communication circuit during the message transmission period. There are two basic categories of error control schemes for data communications: automatic-repeat-request (ARQ) schemes and forward-error-correction (FEC) schemes; in this report, several algorithms of each type are evaluated, as well as several hybrid algorithms in which characteristics of both schemes are employed. The purpose of this study was to determine the feasibility of an error control option for the Group 3 facsimile standard, which would increase the reliability of transmission over the public switched telephone network (PSTN).

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL Dennis Bodson	22b. TELEPHONE (*Include Area Code*) 202-692-2124 / 22c. OFFICE SYMBOL NCS-TS

DD FORM 1473, 84 MAR 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

NCS TECHNICAL INFORMATION BULLETIN 87-4

ERROR CONTROL OPTION FOR GROUP 3 FACSIMILE EQUIPMENT

February 1987

PROJECT OFFICER

DENNIS BODSON
Senior Electronics Engineer
Office of NCS Technology
and Standards

APPROVED FOR PUBLICATION:

Dennis Bodson

DENNIS BODSON
Assistant Manager
Office of NCS Technology
and Standards

FOREWORD

Among the responsibilities assigned to the Office of the Manager, National Communications System, is the management of the Federal Telecommunication Standards Program. Under this program, the NCS, with the assistance of the Federal Telecommunication Standards Committee identifies, develops, and coordinates proposed Federal Standards which either contribute to the interoperability of functionally similar Federal telecommunication systems or to the achievement of a compatible and efficient interface between computer and telecommunication systems. In developing and coordinating these standards, a considerable amount of effort is expended in initiating and pursuing joint standards development efforts with appropriate technical committees of the Electronic Industries Association, the American National Standards Institute, the International Organization for Standardization, and the International Telegraph and Telephone Consultative Committee of the International Telecommunication Union. This Technical Information Bulletin presents an overview of an effort which is contributing to the development of compatible Federal, national, and international standards in the area of facsimile standards. It has been prepared to inform interested Federal activities of the progress of these efforts. Any comments, inputs or statements of requirements which could assist in the advancement of this work are welcome and should be addressed to:

Office of the Manager
National Communications System
ATTN: NCS-TS
Washington, DC 20305
(202) 692-2124

ERROR CONTROL OPTION FOR
GROUP 3 FACSIMILE EQUIPMENT

February, 1987

Final Report

Submitted to:

NATIONAL COMMUNICATIONS SYSTEM

Office of Technology and Standards

Washington, DC 20305

Contracting Agency:

DEFENSE COMMUNICATIONS AGENCY

Contract Number - DCA100-83-C-0047

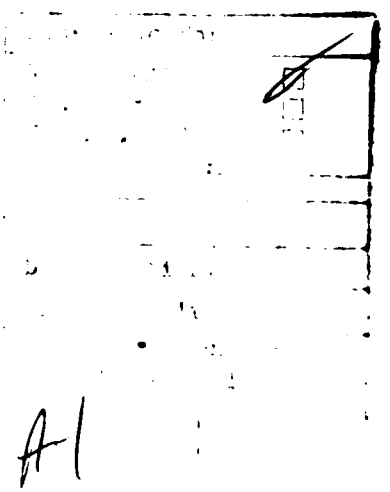
Modification/Task Number - P00010/8

D E L T A I N F O R M A T I O N S Y S T E M S, I N C.

Horsham Business Center, Bldg. 3

300 Welsh Road

Horsham PA 19044



A1

Table of Contents

1.0 INTRODUCTION

This document summarizes work performed by Delta Information Systems, Inc., for the Office of Technology and Standards of the National Communications System, an organization of the U. S. Government, headed by National Communications System Assistant Manager Dennis Bodson. Mr. Bodson is responsible for the management of the Federal Telecommunications Standards Program, which develops telecommunications standards, the use of which is mandatory for all Federal agencies. The purpose of this study, performed under modification number P00007 of contract number DCA100-83-C-0047, was to determine the feasibility of an error control option for the Group 3 facsimile standard, which would increase the reliability of transmission over the public switched telephone network (PSTN).

Group 3 facsimile equipment, which meets CCITT Recommendations T.4 and T.30 (Federal Standards 1062 and 1063, respectively), is now used extensively throughout the world and throughout the U. S. Government. This equipment has no error control capability; the output image quality and transmission bit rate are totally dependent upon the communication circuit during the message transmission period. There are two basic categories of error control schemes for data communications: automatic-repeat-request (ARQ) schemes and forward-error-correction (FEC) schemes; in this report, several algorithms of each type are evaluated, as well as several hybrid algorithms in which characteristics of both schemes are employed.

This report is comprised of four sections. Section 1.0 provides a brief description of the objectives of the study and contains a synopsis of the results obtained and conclusions reached. Section 2.0 contains a discussion of the general considerations made prior to performing the analysis of the error control techniques. The analysis of the error control algorithms is presented in Section 3.0, and a summary of the analysis, including comparison charts and graphs, is presented in Section 4.0.

1.1 Synopsis

Several automatic-repeat-request (ARQ) error control techniques were evaluated in this study, including page retransmission ARQ, stop-and-wait ARQ, go-back-N ARQ, selective-repeat ARQ, and page selective-repeat ARQ. There are two basic types of forward error correction (FEC) techniques, block code FEC and convolutional code FEC. In this study, several examples of each type of FEC error control technique were evaluated, including three block codes (Hamming single error correction/double error correction codes, BCH cyclic codes, and Reed-Solomon cyclic codes) and three convolutional codes (threshold decoding, Viterbi decoding, and sequential decoding).

Two interleaving methods, which are employed to improve performance of the FEC error control techniques, were evaluated; interleaving is a method of distributing a burst of errors among many code words so that the number of errors in each code word is within it's correcting capacity. Several hybrid ARQ-FEC

techniques were evaluated as well, including go-back-N ARQ with FEC, selective-repeat ARQ with FEC, and page selective-repeat ARQ with FEC.

Of the ARQ error control schemes evaluated, the page retransmission ARQ techniques is the simplest and most compatible with existing Group 3 equipment; however, this method has no provision for correcting errors if the retransmit error threshold is not exceeded. The stop-and-wait ARQ technique is the least effective in terms of throughput rate because of the high overhead data required to implement it. The selective-repeat ARQ scheme performs the best in terms of throughput; however, this method requires a full-duplex transmission capability, which is not currently available in Group 3 facsimile and thus would require hardware modifications. The page selective-repeat ARQ error control scheme appears to be the best compromise, in terms of performance and compatibility, among the ARQ techniques. It produces throughput comparable to selective-repeat ARQ without requiring full-duplex operation.

FEC error control techniques could provide essentially error-free document transmission at throughput rates comparable to those achieved by ARQ techniques if data link errors were random and independent. However, actual transmission channels are characterized by a combination of random and burst errors, and FEC error control performance falls off drastically in the presence of burst errors. Interleaving helps the performance of the FEC schemes in the presence of burst errors somewhat, but not

to the degree that they perform better than the ARQ techniques evaluated in this study. The hybrid ARQ error control techniques evaluated provided a marginal improvement in performance over the corresponding ARQ schemes without FEC; however, this improvement does not seem to be of sufficient significance to offset the disadvantages of added complexity and reduced throughput associated with the hybrid ARQ schemes.

2.0 GENERAL CONSIDERATIONS

2.1 Image Size

Typical facsimile page sizes vary from CCITT document #1 Modified READ coded at 3.85 l/mm (1728 pels/line x 1168 lines with 24.8 compression ≅ 80,000 bits) to CCITT document #7 Modified HUFFMAN coded at 7.7 l/mm (1728 pels/line x 2336 lines with 4.88 compression ≅ 830,000 bits). Even larger coded images may result if larger document widths or lengths are selected.

For purposes of calculating comparative throughput rates for various error correcting schemes, CCITT document #1, Modified HUFFMAN coded at 7.7 l/mm, will be used to represent the typical document. In compressing this document, 15:1 compression is achieved, resulting in a coded image size of approximately 270,000 bits.

2.2 Modem Considerations

The Rockwell V96 P/1 is a typical, widely used fax modem. It has full duplex capability, but only for a 4-wire system. Half duplex operation only is available for use with the 2-wire public switched telephone network (PSTN). This implies, of course, that the full duplex go-back-N ARQ or selective-repeat ARQ re-transmission protocols cannot be used without modifying the modem to provide a back channel for the acknowledge response.

All high speed data transfers to the modem receiver must be preceeded by a training sequence of approximately 250 ms duration for 9600/7200 bps data rates and of approximately 1000 ms duration

for 4800/2400 bps data rates. This represents overhead which tends to accentuate the inherent slowness of the stop-and-wait ARQ error control technique. The Rockwell modem requires a full train when line connection is first made, but permits use of a shorter 'retrain' after that, thus reducing the overhead somewhat. This retrain is not part of Federal Standard 1063 (EIA Standard RS-466), however.

Any ARQ acknowledge message should probably use the 300 bps FSK mode of operation for three reasons:

a) it is compatible with existing Group 3 procedures;
b) when the training sequence time of the high speed mode is taken into account, a short 300 bps HDLC (high level data link control) message requires no more time than the same message using the high speed mode;
c) the 300 bps error rate is much lower (4 to 14db), an important consideration since the data transmitter must assume that any response message containing an error is a NACK and will retransmit the block with which it was associated.

There are two aspects of Group 3 facsimile data transmission which affect error patterns.

a) The modulation method codes 2, 3, or 4 bits in the phase and/or amplitude of one signal element. Any noise which corrupts the element tends to affect all of the bits coded in it.

b) The data is scrambled with a 7-bit polynomial (V27) or a 24-bit polynomial (V29) which extends bursts of errors by the polynomial length and multiplies random errors by 3.

Both of these tend to accentuate the burst-like nature of real line errors, thus making the assumption of a random independent bit error environment even less valid than it would be otherwise (See Section 2.5).

Some of the ARQ techniques analyzed in this study require the addition of an acknowledge message back channel to the modem for full duplex handshaking on the PSTN. Assuming a 4K bit data block (See Section 2.4), the worst case (9600 bps) data block transmission time is approximately 430 ms. The HDLC acknowledge message is about 10 bytes long, so the minimum acknowledge bit rate is about 200 bps. The already implemented (but not as a back channel) 300 bps FSK channel seems the logical choice for the back channel.

2.3 Data Block Format

The HDLC frame structure seems the logical choice for the data block format. It is already part of the Group 3 FAX protocol, it is a bit oriented protocol, and the 16-bit CRC error control field is adequate for any reasonable choice of block length. Figure 2.1 shows a typical HDLC data block format.

The information field consists of an 8-bit or 16-bit block identifier field, which contains the number (1 to N) of the block

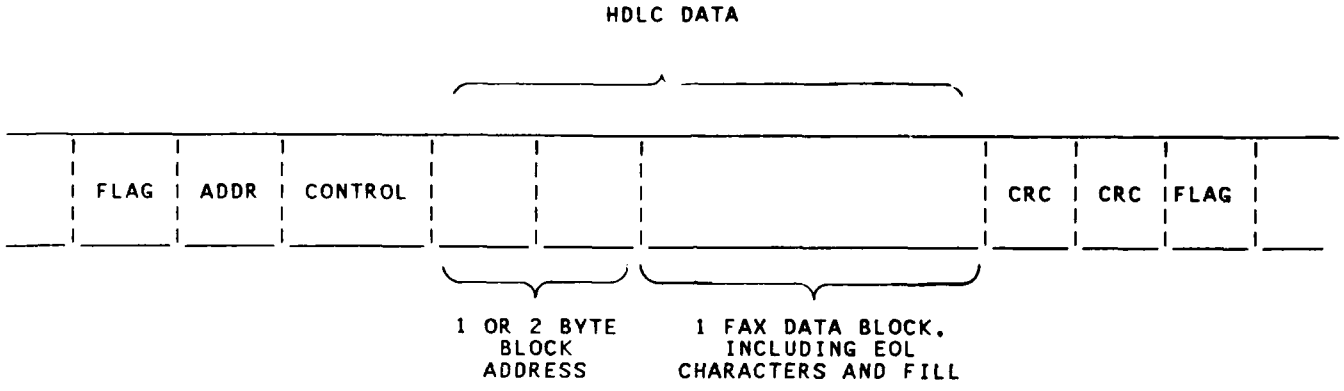


FIGURE 2.1 HDLC DATA BLOCK FORMAT

being transmitted, and a facsimile data field.

The control field can have the same format used in a normal Group 3 FAX message. Bit 5 = 1, which indicates the final frame within a procedure, is redundant with the control message preceeding the data, and might be used to indicate the last data block in a page.

The last data block can be filled out with trailing zeroes to make it conform with the standard block length. This might facilitate receive operations in some machines at a slight cost in throughput.

The standard HDLC 16-bit CRC check field detects all error patterns not divisible by the 16-bit generator polynomial. These include all patterns with an odd number of bits in error, all 2-bit errors, all error bursts spanning 16 or fewer bits, and most longer bursts, all provided the data block length is less than $2^{15}-1$ (about 32K) bits. Shorter blocks have an even lower undetected block error rate. It is possible that single bit errors which corrupt a leading or trailing FLAG or cause a false FLAG in the data field can cause undetectable frame errors, but these cases have little real effect on performance.

The probability of any error pattern being divisible (undetected) by the 16-bit polynomial is approximately 1.5×10^{-5}. It has been estimated (ref. 1) that, with a block length of 800 bits, the probability of an undetected block error is 10^{-8}, a truly negligible number.

The HDLC protocol overhead is approximately 10 bytes (80 bits) per frame. For a 4000 bit data block, this and the bits added to the data field following five consecutive 1's reduce the throughput by a factor of about 0.96. This is common to all error correction methods analyzed herein, and is generally neglected in throughput calculations.

2.4 Optimum Data Block Size

The HDLC data block size for ARQ operation should be selected to minimize the average document transmission time. The block should be large enough to minimize the transmission protocol and response delay overhead but small enough to make re-transmission efficient and to achieve a high probability of successful retransmission.

The optimum stop-and-wait ARQ block tends to be larger than the optimum continuous transfer block because the larger block reduces the response overhead.

For either stop-and-go or continuous transmission strategies, a higher error rate makes smaller blocks more efficient because the amount of wasted retransmission of correct data is reduced.

The optimum block size is reduced if the last transmitted block is terminated with the last real data rather than filled out with zeroes. The effect of this is minimized, of course, as the number of blocks per message increases.

Chu (ref. 2) analyzes the problem for stop-and-wait ARQ, go-back-N ARQ, and selective-repeat ARQ, and for both random and

burst errors. He assumes geometric message length distribution, and obtains expressions for total message transmission times, minimizing them as a function of block length.

Chu's average transmission time is minimum when:

$$(q^{-B}-1)\left[\,1 + (AR+B+b)\left(\frac{K}{1-K(B+b)}\right)\right] + [AR+B+b]\ \ln q = 0$$

Where B = data bits per block

b = overhead bits per block

A = stop-and-wait ARQ response delay

(A = 0 if continuous operation)

R = bit rate

K = bit error rate

q = 1 - p such that

$\tilde{L}$ = average message length = $\frac{1}{p}$ and

(p + q) = 1

This equation is difficult to evaluate for our specific parameters. Information taken from plotted results indicate:

a) Stop-and-wait ARQ

1. For random errors and assuming a 1 second response delay, the optimum block size is approximately 4000 bits at $K = 10^{-4}$ and approaches the message length at $K = 10^{-6}$.
2. Under burst error conditions the block size approaches the message length.

b) Continuous transmission ARQ

1. For random errors, the optimum block length is approximately 500 bits at $K = 10^{-4}$ and 5000 bits at $K = 10^{-6}$,

relatively independent of whether the go-back-N or selective-repeat technique is used.

2. For burst errors, the optimum block length is 3000 to 4000 bits at 4800 bps, and nearer 8000 bits at 2400 bps.

Based on this data, a 4000 bit block (512 bytes) seems a good compromise. This requires a 2-byte block number of field to accomodate the number of blocks in very large (low compression) documents.

There is little to be gained in attempting more exact calculations of optimum block length because of the uncertain transmission error characteristics.

2.5 Line Noise Characterization

Transmission channels inevitably add noise to the transmitted signal. Often the noise has a Gaussian amplitude distribution and a flat (white) frequency spectrum, but it also usually includes interference with a strongly non-Gaussian amplitude distribution caused by impulses, fading, and the like.

On a binary, symmetric transmission channel the Gaussian noise produces random independent bit errors which can be represented by a bit error probability or rate. The impulse noise produces bursts of errors.

Most analyses of error control systems, including those in this study, are based on random independent bit errors because these are easy to treat mathematically. This is a great over-simplification, however, as can be seen from the data

presented in this section. Burst errors are a significant part of real error patterns.

The modem modulation techniques and data scramblers specified in Federal Standard 1062 (EIA Standard RS-465) also tend to produce error bursts, as described in Section 2.2.

ARQ error control techniques are much less sensitive to error patterns than FEC techniques. An ARQ system cares only that a block contains an error, not how many errors it contains or what the error pattern is. FEC, on the other hand, has maximum correction capability and minimum complexity with random errors. Performance degrades rapidly in the presence of burst errors. Generally, ARQ error detection is much more reliable than FEC error correction.

There seems to be little data available on line error patterns at 9600 bps. The results of the 1969-1970 AT&T connection survey for 4800 bps data (ref. 1) indicate that error patterns are very non-random. At 4800 bps, 40% of calls had a burst error rate $>2\times10^{-6}$, a bit error rate $>10^{-5}$, and a 1K block error rate $>10^{-5}$. At 3600 bps with a guard space of 50 bits, 10% of error bursts were 70 bits or more in length while 30% were 25 bits or more.

The connection study reported in the November 1984 Bell Telephone Technical Journal indicated that at 4800 bps the 1K bit block error rate was 10^{-3} or less for 70% of long line connections and for 85% of short connections.

2.6 Impact on Existing Group 3 Protocol

Generally, the addition of error control to Group 3 facsimile will have little affect on the protocol outside of the data transfer itself. The one exception to this is the page selective-repeat scheme proposed by the United Kingdom in COM VIII-13-E, which requires the addition of a partial page request (PPR) message requesting retransmission of error data blocks.

The error control must be optional to maintain compatibility with existing Group 3 equipments, and one of the reserved bits in the Facsimile Information Field of DIS, DTC, DCS messages can be assigned to this function (again, as in COM VIII-13-E).

The stop-and-wait, go-back-N, and selective-repeat ARQ systems all require data block acknowledge messages, which will be in the standard 300 bps FSK HDLC format with a FIF consisting of a one or two byte block address and a one byte ACK or NACK character. A timeout must be implemented for these messages, and some operation defined for lack of a valid response in the timeout period. One possible sequence after a timeout is:

a) In a stop-and-wait ARQ system, the preceeding block is retransmitted. After 3 consecutime timeouts, the transmitter disconnects.
b) In go-back-N or selective-repeat ARQ systems, the transmitter returns to the earliest unacknowledged data block and continues transmission from that point. After 3 consecutive timeouts the transmitter disconnects.

There must be some limit on the number of block re-transmissions. If this limit is reached, the transmitter could disconnect or could enter a mode where NACK responses are ignored and the remainder of the data blocks are transmitted unconditionally. The receiver should always be prepared to accept this mode of operation.

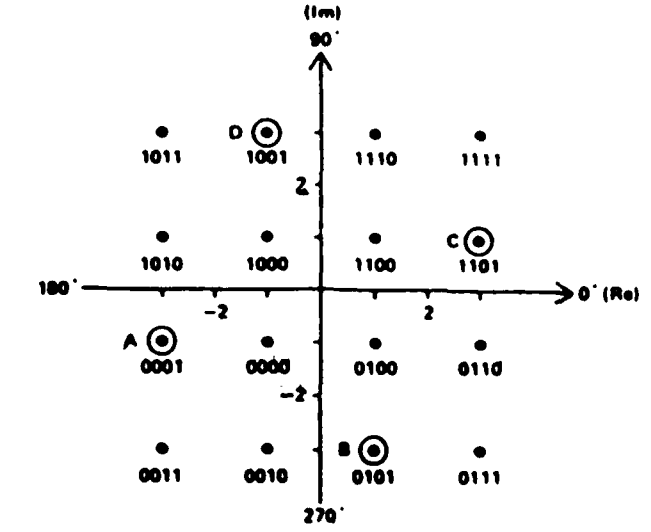
Existing Group 3 facsimile error detection (incorrect line lengths, missing EOL's) and correction (white line or repeat previous lines) should remain in place to accomodate the occasional error that is missed by ARQ and very noisy conditions where ARQ is ineffective.

2.7 CCITT Recommendation V.32 Error Correction

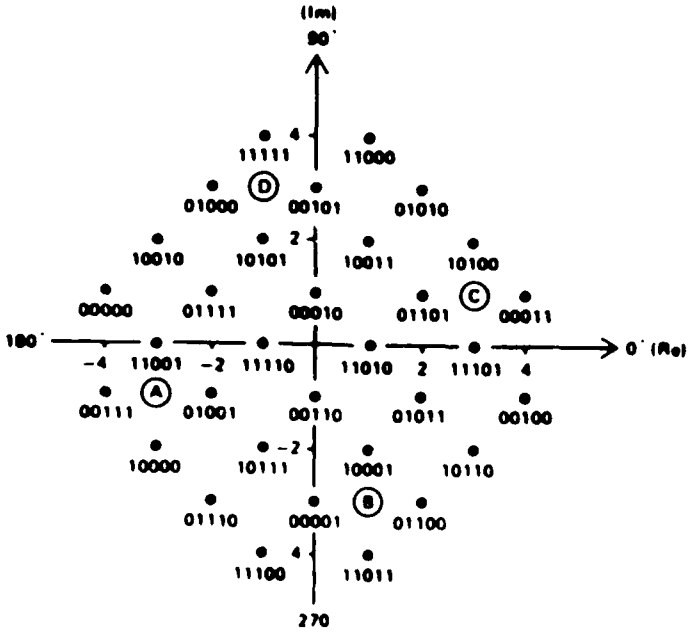
CCITT Fascicle VIII.1, Recommendation V.32 describes a family of modems which have an optional trellis coding error correction mode at 9600 bits/s.

Operation without error correction is similar but not identical to that defined in Federal Standard 1062 (EIA Standard RS-465). Data is divided into 4-bit groups which are coded into 16 phase-amplitude points (not the same points as in Federal Standard 1062) in a single modulation interval, as shown in Figure 2.2(a).

The optional error correction mode encodes this 4 bit group into a 5 bit convolutional code, which is then encoded into 32 phase-amplitude points in the same modulation interval, as shown in Figure 2.2(b). The receiving modem uses error



(a)



(b)

FIGURE 2.2 CCITT RECOMMENDATION V.32 MODEM CODING

correcting/detecting trellis decoding of these 5 bits to recover the 4 data bits.

This method has the advantage of adding error correction without reducing the data rate. It's obvious disadvantages are:

a) the phase-amplitude points representing the 5-bit groups are closer together than those for the 4-bit groups and are therefore more susceptible to noise errors; and

b) more elaborate modems are required.

Performance data for these modems in the real PSTN environment is apparently not yet available.

3.0 ANALYSIS OF ERROR CONTROL ALTERNATIVES

Section 3.1 analyzes various re-transmission or automatic-repeat-request (ARQ) techniques. Most of these involve transmitting the fax data in high level data link control (HDLC) blocks using the CRC frame check field for error detection. Section 3.2 examines the use of coding to perform forward error control (FEC) as an alternative to ARQ. Section 3.3 analyzes some hybrid schemes, which use both ARQ and FEC techniques.

Throughput rate is used in this study as the primary indicator of system performance. All ARQ throughput calculations are approximate, assuming a noiseless feedback channel and perfect error detection. Uncertainty in transmission error characterization and ARQ response times make any attempt at more exact calculations meaningless. The throughput calculations are based only on actual data transfer times - they do not include link and circuit establishment, disconnect procedures, etc.

3.1 ARQ Techniques

3.1.1 Full Page Retransmission

The simplest error control method is already implicit in Group 3 facsimile; if the quality of received document is unacceptable, the receiver can respond with a PIN message, requesting operator intervention with the possibility of re-transmission of the document. This could be formalized and automated in the following manner:

a) A page of data is transmitted in the existing Group 3

format. The transmitter either stores the full page as it is transmitted or else has provisions for automatic document re-scan.

b) The receiver evaluates document quality using an algorithm based on the percentage of lines correctly decoded. If quality is acceptable, the receiver responds as is appropriate. If the quality was unacceptable, the receiver responds with a new message requesting re-transmission.

c) Upon receipt of the re-transmission request, the transmitter sends a DCS/TCF message which reduces the bit rate to either the next lower value or unconditionally to 2400 bps, and then re-transmit the document.

d) After three re-transmissions, the transmitter would arbitrarily choose whether to disconnect, re-transmit the document again, or continue with the fax procedure.

To calculate the throughput for this page repeat method, consider each document line to be a 'block'. CCITT Doc. #1, Huffman coded, has 1168 lines coded in 270,000 bits, an average of 231 bits per line. The probability of a line error is approximately:

$$P_L = [1 - (1 - P_E)^{231}]$$

where P_E is the bit error rate.

Throughput for other methods in this section are calculated at 4K bit block error rates (P_B) of 10^{-1} and 10^{-2}. These correspond to bit error rates of 2.5×10^{-5} and 2.5×10^{-6}, respectively.

P_B	P_E	P_L
10^{-2}	2.5×10^{-5}	1.6×10^{-3}
10^{-1}	2.5×10^{-6}	6.0×10^{-3}

With a 1% line error retransmit criteria, the retransmit probability is approximately

$$P_R \cong [1-(1-PL)^{100}] = .148 \text{ at } P_B = 10^{-2}$$
$$= .452 \text{ at } P_B = 10^{-1}$$

Neglecting response time, throughput is approximately

$$T = \frac{B}{B(1 + \frac{P_R}{1-P_R})} = 1 - P_R$$

$$T = .85 \text{ at } P_B = 10^{-2}$$
$$= .55 \text{ at } P_B = 10^{-1}$$

These figures are not really comparable to other error control methods because no errors are corrected if the retransmit error level is not exceeded.

3.1.2 Stop-And-Wait ARQ

In a stop-and-wait ARQ system, the transmitter sends a block of data and then stops, waiting for a positive (ACK) or negative (NACK) acknowledgement from the receiver before transmitting the next block or re-transmitting the same block.

The data block format would be the HDLC frame discussed in Section 2.3. The receiver would use the HDLC CRC check code, which detects essentially all transmission errors, as the primary criteria for making an ACK/NACK response decision. It could also use any other information at it's disposal, such as soft error or loss of carrier flags from the receiver modem or run-length decoding errors. The response should be a standard Group 3 fax 300 bps FSK HDLC message with a one byte data field containing the ACK or NACK character. There are three reasons for this choice: it is compatible with existing Group 3 procedures; a high speed response might require a train sequence, which would negate the gain in speed; and the 300 bps FSK error rate is many orders of magnitude better than high speed error rates.

Although theoretically not necessary, it might be useful to identify each transmitted data block with the block number field discussed in Section 2.3. Under some circumstances, such as repeated NACK responses, the transmitter might proceed to the next data block unconditionally. The block number would then inform the receiver that the last block received must be accepted even though it contained errors. The Group 3 fax protocol outside of the data transmission is unchanged.

Part of the response overhead of the stop-and-wait ARQ system is the modem training sequence which might be required preceding each data block. The duration of this sequence as specified in Federal Standard 1063 (EIA Standard RS-466) is 250 ms for 7200/9600 bps operation and 1000 ms for 2400/4800 bps operation.

Both the transmitter and receiver need store only one block of data. The throughput of a stop-and-wait ARQ system is approximately given by:

$$T = \frac{B}{(B + AR)\ 1 + \frac{P_B}{1-P_B}} = \frac{B\ (1 - P_B)}{B + AR}$$

where

B = number of bits per block

A = response delay (including modem train time)

R = signaling rate (2400 to 9600 bps)

P_B= block error rate

$\frac{P_B}{1 - P_B} = \sum_{i=1}^{\infty} [P_B]^i$ = Probable number of re-transmissions of a block

Assuming a 1 second response delay, 9600 bps signaling rate, 4000 bit blocks, and no errors:

$$T = 0.29 ;$$

assuming a 10^{-1} block error rate:

$$T = 0.26 .$$

It is obvious that the performance of the stop-and-wait ARQ system is degraded not so much by the block error rate as it is by the response overhead.

3.1.3 Go-Back-N ARQ

In a go-back-N ARQ system, the transmitter continuously transmits data blocks and stores them pending receipt of an acknowledgement for each. If a negative acknowledgement for a particular block is received, the transmitter returns to that block

(after finishing the current block to maintain synchronization), re-transmits it, and continues transmitting from that point, including re-transmission of all blocks sent in the intervening time between the original error transmission and receipt of the negative acknowledgement.

This method is much more efficient than the stop-and-wait ARQ, since it eliminates the idle "wait" overhead following a successful block transmission.

Since data transmission is continuous, only a start-of-message modem train is required. It is necessary, however, to transmit N dummy data blocks at the end of each message in order to maintain data continuity pending receipt of the last acknowledgement.

The go-back-N ARQ requires full duplex modem operation. This is a major disadvantage since it is not normally available on modems interfacing to the 2-wire public switched telephone network (PSTN). Modification of existing modem design to provide a back channel - most probably utilizing the 300 bps FSK signaling now used for Group 3 control messages - would be required to implement this system.

Existing Group 3 fax protocol outside the data transmission function is unchanged. Some protocol for handling erroneous or missing acknowledge messages is needed - probably retransmission beginning at the block in question.

Although some published descriptions of go-back-N ARQ systems postulate a fixed N block response delay, this is not practical for use with Group 3 fax, which operates at bit rates of 2400 to 9600

bps over a wide variety of line lengths. Real values of N might vary from 2 (almost immediate response) to about 8 (3 second response delay with 4K bit blocks at 9600 bps). It is desirable, therefore, to have a block identifier tag in both the data block and in the acknowledge messages to facilitate acknowledge handshaking. The data block would have the HDLC format described in Section 2.3 and the acknowledge message would be an HDLC frame with a two or three byte data field; a one or two byte block identifier and the ACK/NACK character.

The acknowledge message transmission time must not exceed the data block transmission time. Assuming a 300 bps FSK acknowledge channel, a 10 byte HDLC acknowledge frame would require about 270 ms. This is compatible with a 4K bit data block, which requires about 430 ms to transmit at 9600 bps.

The throughput of a go-back-N ARQ system is approximately given by:

$$T = \frac{B}{B\left(1 + \frac{P_B(N+1)}{1 - P_B}\right)} = \frac{1 - P_B}{1 + NP_B}$$

where B = number of bits per block

N = number of blocks returned after negative acknowledge. (N+1) = number of blocks retransmitted.

P_B = block error rate

$$\frac{P_B}{1-P_B} = \sum_{i=1}^{\infty} [P_B]^i = \text{expected number of retransmissions per block.}$$

The number of blocks returned N is determined as follows:

$$N = (1 + \frac{DR}{B})$$

where D = round trip acknowledge delay,

B = number of bits per block,

R = data transmission rate.

Assuming D = 1 second, B = 4000 bits, and R = 9600 bps, then N = 3 and

and

$$T = \frac{1 - 0.1}{1 + 0.3} = 0.69 \text{ for } P_B = 10^{-1}$$

$$T = \frac{1 - 0.01}{1 + 0.03} = 0.96 \text{ for } P_B = 10^{-2}$$

As expected, this throughput is much higher than that of the stop-and-wait ARQ system, and is much more dependent upon block error rate.

The go-back-N ARQ transmitter must be capable of storing enough data blocks to accomodate the worst case response delay, approximately 8 4K bit blocks = 4K bytes. The receiver must store enough blocks to accomodate it's worst case decoding delay timer. Assuming that block N is decoded during receipt of block N+1, the receiver needs minimum of two blocks of data storage.

3.1.4 Selective-Repeat ARQ

In a selective-repeat ARQ system the transmitter sends data blocks continuously in order and also stores them pending receipt of an ACK/NACK for each. When the transmitter receives an NACK response, it finishes the current transmission (block N), re-transmits the error block, and then resumes normal transmission with block N+1. Data transmission is continuous, so a modem train sequence is necessary only at the beginning of the message.

The data blocks and response messages would use the same HDLC formats described in Section 2.3. Both would have block identifier fields.

This method has the same full-duplex line requirement as the go-back-N ARQ technique, with the associated modem hardware disadvantages. It requires more receiver data storage - enough to accomodate worst case response delay - but yields slightly higher throughput rates.

The expression for selective-repeat throughput rate is

$$T = \frac{B}{B\left(1+\frac{P_B}{1-P_B}\right)} = 1-P_B$$

where P_B = the block error rate

B = the number of bits per block.

$$\frac{P_B}{1-P_B} = \text{the probability of block retransmission}$$

For a block error rate = 10^{-1} ,

$T = 0.90$;

for a block error rate = 10^{-2} ,

$T = 0.99$.

3.1.5 Page Selective-Repeat ARQ

This is the method proposed by the United Kingdom in COM VIII-13-E. An entire page of data is transmitted in HDLC blocks. The receiver checks each block, and, when the page is complete, either accepts it with a normal Group 3 response if it contains no errors, or requests retransmission of the erroneous blocks (from 1 block to the entire page) with a special response identifying those blocks.

The error blocks are then re-transmitted (probably, but not necessarily, in numerical order) in the same manner as a normal fax data transmission, preceded by a modem train sequence. The response/retransmission sequence continues until the receiver assembles an error free page.

This approach is a compromise - it achieves a higher throughput than the stop-and-wait ARQ method without the full duplex modem operation required by the continuous transmission ARQ methods. It does, however, require both transmitter and receiver to store a full page of data.

Existing Group 3 protocol is changed less for this method than

for the go-back-N and selective-repeat ARQ methods; the single re-transmission request message is simpler than individual block acknowledges, which can present control problems if response timeouts occur.

One disadvantage of this method is that printing may be delayed: the receiver cannot proceed printing past an error block until the block is corrected, and while the go-back-N and selective-repeat methods correct a block almost immediately, the page selective-repeat must wait for transmission of the remainder of the page before correction occurs.

The probable number of retransmissions of a block is:

$$\sum_{i=1}^{\infty} [P_B]^i = \frac{P_B}{1-P_B}$$

where P_B is the block error probability.
The total number of data bits transmitted is, therefore,

$$HB\ (1 + \frac{P_B}{1-P_B})$$

where H is the total number of data blocks in the original page and B is the number of bits per block.

The interpage delay = AR bits where A is the delay in seconds and R is the transmission rate.

The probability of an X block page containing no errors is $(1 - P_B)^X$. The total number of page re-transmissions is therefore given by:

$$\sum_{i=0}^{\infty} \left[1 - (1-P_B)^{(P_B H^i)}\right]$$

where $P_B H^i$ is the number of blocks in the retransmitted page (error blocks).

The expression for throughput is then

$$T = \frac{\text{bits in original page}}{\text{total bits transferred + total inter page delay (bits).}}$$

$$T = \frac{HB}{HB\left(1 + \frac{P_B}{1-P_B}\right) + AR \sum_{i=0}^{\infty} \left[1-(1-P_B)^{(P_B H^i)}\right]}$$

Assuming B = 4K bits per block

H = 70 blocks (CCITT Doc. #1, Huffman coded)

R = 9600 bps

$P_B = 10^{-1}$

Then T = 0.82 for A = 1 second.

T = 0.78 for A = 3 seconds.

If $P_B = 10^{-2}$,

T = 0.95 for A = 1 second.

T = 0.93 for A = 3 seconds.

As expected, these results are much better than those achieved by the stop-and-wait ARQ method, but not quite as good as those for the selective-repeat ARQ method. Comparison with the go-back-N ARQ throughput is interesting; at a block error rate of 10^{-2} the go-back-N method is slightly faster, but at a block error rate of 10^{-1} the burden of transmitting the extra N blocks after each error

block is greater than the inter-page delay overhead, and go-back-N method is significantly slower.

3.2 Forward Error Correction

Forward error correction (FEC) can be considered an alternative to the ARQ techniques described above. An FEC system transmits redundant check bits along with data bits; even though noise causes errors in both data and check bits, in most cases there is enough information remaining in the code for the receiver to correct the errors and recover the data.

The throughput of an FEC system is constant and is equal to the code rate. If k data bits are transmitted in an N bit code block with (N-k) redundant bits, the rate is equal to k/N. In general, the lower the code rate the better the error detecting and correcting performance of the code.

In a pure FEC system, we are concerned only with the error correcting capacity of a code, since there is no re-transmission to correct an error that is only detected. In hybrid systems (Section 3.3), both are important.

There are two types of FEC codes: block codes, in which the (N-k) check bits in an N bit code word apply only to the K data bits in that word, and convolutional codes, in which the check bits also apply to data in preceding code words.

3.2.1 Block Codes

An (n,k) block code adds (n-k) check bits to a k bit data block

to form an n bit code word. Each block code word is independent of all other code words.

A block code is linear if it has a certain structure imposed on it. Its members are linear combinations of a set of linearly independent rows (vectors) in a generator matrix. Most mathematical techniques for analysis and synthesis of codes depend upon this structure, and most known codes which are effective are linear.

A block code is systematic if it contains the generating bits unmodified; i.e., if it is formed by appending (n-k) check bits to the k data bits. Any non-systematic linear block code is equivalent to a systematic linear code.

If two code words differ in position (bits), then the 'distance' between these two words is d. The minimum distance of a code is the minimum value of d between any two code words.

$$d_{min} \leq 1 + (n-k) \ .$$

In a perfect code, all code words have the same distance from each other, and

$$d_{min} = 1 + (n-k) \ .$$

Cyclic block codes have the property that every cyclic shift of a code word is another code word. This simplifies coding and decoding hardware.

The receiver decoder calculates a 'syndrome' based on parity checks of various fields of data and check bits. A non-zero syndrome identifies a transmission error, and the syndrome also identifies the bit(s) in error if the correction capacity of the

code has not been exceeded.

The error detection (d) and correction (c) capacities of a code are:

$$c \leq \frac{d_{min} - 1}{2} \quad \text{and}$$

$$d \leq d_{min} - 1$$

These can't be achieved simultaneously.

There are many good block codes available:

a) Hamming single error correction/double error detection codes are available from many manufacturers in single integrated circuit packages with codes varying from the basic (7,4) to (80,72).

b) BCH cyclic codes are the best known binary block codes for correcting random errors. Many good BCH codes are known, with lengths up to hundreds of bits.

c) Reed-Solomon (RS) codes are non-binary (i.e. character rather than bit symbols) cyclic codes which have maximum distance characteristics ($d_{min} = n-k+1$). They are a subset of BCH codes. Reed-Solomon codes are widely used; for example, compact audio disc equipment uses (32,28) and (28,24) RS codes, and a (31,15) RS code is a JTIDS standard for tactical military communications systems. RS codes are not directly applicable to binary fax data, but could be used as part of a compound code in conjunction with a short internal binary code.

3.2.2 Convolutional Codes

Convolutional codes divide a data stream into (n,k) frames with k data bits and (n-k) check bits, just as block codes do. Data length k is typically much shorter than that of block codes, however; often, k = 1. The check bits are a function not only of the current data bit(s), but also of the data in previous frames.

The general form of a 1/n convolutional encoder is shown in Figure 3.1(a). A typical (rate = 1/3) encoder is shown in Figure 3.1(b).

Useful parameters in evaluating convolutional codes are code rate and constraint length:

Code rate R = k/n bits/frame

Constraint length C = nK, where

n = bits per code frame

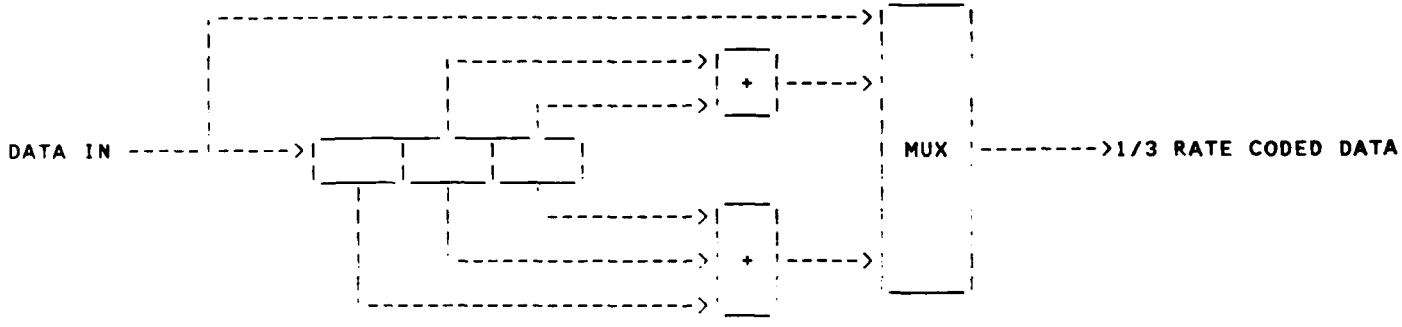
K = bits in generating shift register.

Constraint length is a measure of how many previous code bits affect the current frame. It can be considered the 'blocklength' of the code.

Although the convolutional codeword is really of infinite length, decoding decisions are necessarily based on finite length segments of the code - at least the constraint length. Generally, the longer the decoding 'window' the better the decoder performance.

There are three principle methods of decoding convolutional codes.

(a)



(b)

FIGURE 3.1 CONVOLUTIONAL CODERS

a) Threshold decoding involves calculating a syndrome in a manner similar to that used for block codes. This method has a higher decoding error probability than that of either Viterbi or sequential decoding in random error environment, but is considered preferable for burst error environments.

b) Viterbi decoding is based on finding the most likely path through the 'trellis' representation of a convolutional code. It's use is confined to short constraint lengths since hardware and speed requirements grow exponentially. It is usually not recommended for use in burst noise environments, because it tends to produce decoder error bursts of several times the input burst length. Viterbi decoding is widely used, and reasonably priced hardware is available. One single package rate 1/2 convolutional coder/Viterbi decoder provides 3 db coding gain and is available for less than $500.

c) Sequential decoding also searches for the most likely path through a message 'tree' or 'trellis'. It's output bit error rate decreases exponentially with constraint length, but it has the disadvantage of rapid increase in search length (and therefore computation time) with increased bit error rate. Like the Viterbi decoder, it is not recommended for burst error correction.

High rate convolutional codes are available, but they require longer constraint lengths and commensurately more hardware to achieve good results. For example, a 7/8 rate convolutional codec

with threshold decoding and a constraint length of 376 bits is available as a two printed circuit board set.

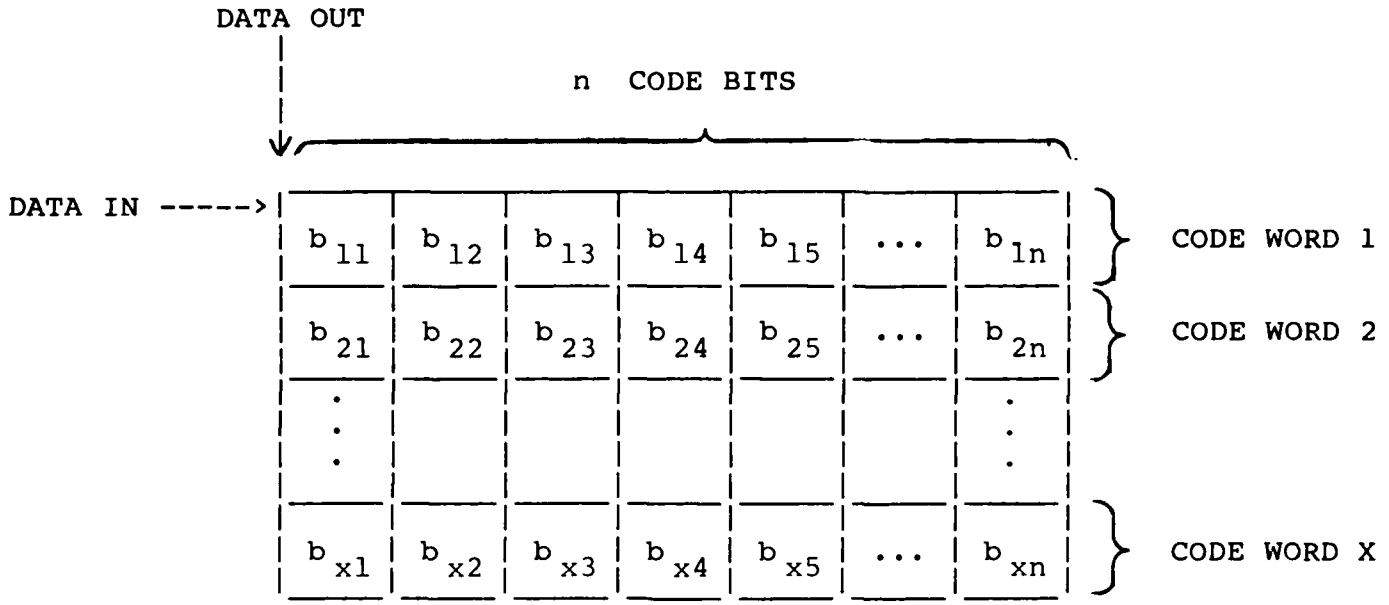
Convolutional codes can be made to cope with noise bursts by adding delay between stages of their generator shift registers, a technique very similar to interleaving block codes. The guard space requirements are fixed between bursts, and are comparable with interleaved block codes.

3.2.3 Interleaving

Interleaving is a technique of distributing a burst of errors among many code words so that the number of errors in each word is within its correcting capacity.

Figure 3.2 illustrates one typical interleaving technique. M words of an (n,k) code are stored in the rows of a memory matrix, and the data is then transmitted by scanning the columns. This essentially converts an (n,k) code with error correcting capability of $t \leq (n-k)/2$, into an (mn, mk) code which can correct a burst of maximum length mt, or some (but not all) combinations of shorter bursts totaling that number. There must, however, be no additional bit errors in the matrix. This means that a guard space of error-free bits between error bursts is necessary to correct the bursts. The guard space varies from m(n-t) bits (error at start start of matrix) to zero bits (error at end of matrix).

Figure 3.3 illustrates another interleaving method, one closely related to convolutional coding techniques. This is the method used in audio compact disk equipment. The bits in an (n,k) code are



INTERLEVING AN (n,k) BLOCK CODE

Data Loaded into rows, unloaded from columns, left to right.

FIGURE 3.2 INTERLEAVE MEMORY FORMAT

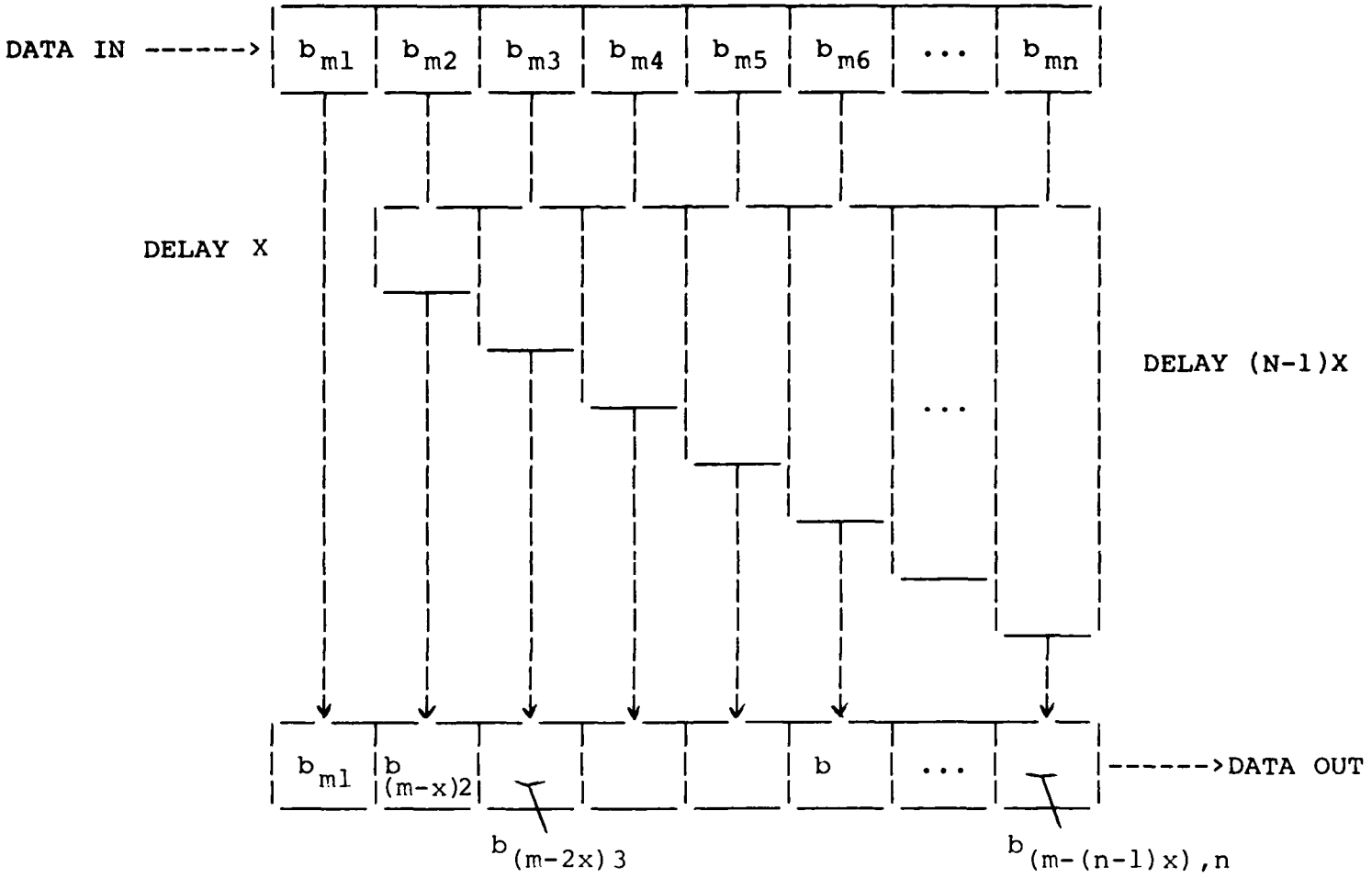


FIGURE 3.3 ALTERNATE INTERLEAVING SCHEME

delayed from zero to x(n-1) bits in x bit increments. The output contains bits from n different words out of a group of xn words. The burst correction performance is the same as the first interleaving method with a = n: if the code can correct $t \leq (n-k)/2$ errors input word, then the interleaving permits a maximum burst of xnt bits to be corrected. The guard space between maximum length bursts here is approximately xn(n-t) bits, which is twice the average guard space required by the first method. This method does, however, allow correction of shorter bursts which follow a maximum burst by fewer guard bits than another maximum burst would require. Both methods continue to correct random errors.

These interleaving methods can be applied to convolutional codes as well as block codes. Convolutional codes can also be designed to accomodate error bursts by using similar techniques; delay is added between generator shift register stages so that, at the decoder, the burst is spread over many code words.

An example of interleaving is as follows:

A Hamming (7,4) code corrects one error.

$$t = 1 \leq \frac{(7-4)}{2}$$

If it is interleaved using the second method with x = 2, maximum bursts of xnt = 14 bits (bursts covering 2 words) or all 13 bit bursts (bursts covering 3 words) are distributed so that only one error bit occurs in each output word. The guard space between maximim bursts is xnt(n-t) = 84 bits, but a one bit error can follow the burst after a guard of only 6 bits.

3.2.4 FEC Performance

The performance of several typical codes in a random error environment are calculated below:

1. Hamming (15,11) single error correcting block code.

The code rate is 11/15, so the typical document of 270,000 bits (see Section 2.1) requires 15/11 (270,000) = 368,000 bits to transmit.

Without coding, one would expect 270 errors in the received document at a bit error rate (BER) of 10^{-3}.

The coded page requires 24,545 15-bit code blocks. The probability of zero or 1 errors per block (no decoded bit errors) is:

$$P_C = \sum_{x=0}^{1} \binom{x}{n} P^x (1-P)^{n-x}$$

where

binary coefficient $\binom{x}{n}$ is the number of size x subsets of n = $\frac{n!}{(n-x)!x!}$

P = bit error rate = 10-3

N = code length = 15

$P_C = (1-P)^{15} + 15P(1-P)^{14}$

$= 0.99989$

The code block error rate is $(1-P_C) \cong 1 \times 10^{-4}$.

Since there are about 25,000 code blocks per page, there will

be approximately 2.5 errors per page, a reduction in transmission error rate of 10^{-2}. The throughput is reduced to 11/15 = 0.73.

2. 1/2 rate convolutional code with Viterbi decoding.

This commercially available single module codec will reduce a BER of 10^{-3} to 10^{-5}, about the same as example 1. The throughput achieved is 0.5.

3. BCH (63,51) two error correcting block code.

There are 4285 63-bit blocks in the reference document.

The probability of 0, 1 or 2 bit errors in a block is :

$$P_C = (1-P)^{63} + 63(P)(1-P)^{62} + [(63)(62/2)](P)^2(1-P)^{61}$$

$$P_C = .999962$$

The block error rate is $(1-P_C) = 3.8 \times 10^{-5}$, which results in an error rate of about 1 every 6 pages, a reduction of approximately 6×10^{-4}. The throughput achieved is 51/63 = 0.81.

If the bit errors on telecommunication channels were really random and independent, these examples would indicate that essentially error free data transmission could be achieved using FEC alone with reasonable length codes at throughput rates comparable to those achieved with ARQ techniques.

Unfortunately, real channels exibit a combination of random and burst errors that make the use of such simple codes of little real value in reducing block error rate.

Interleaving (see Section 3.2.3) can improve the burst

performance of FEC error control techniques. For example, interleaving the (63,51) BCH code 5 deep increases the correctable burst length from 2 to 10 bits. It also requires, however, that an error free guard space with an average length of 150 bits (305 bits maximum) must follow such a burst. Statistics indicate that the addition of interleaving still leaves a significant percentage of error blocks uncorrectable.

Convolutional codes have no intrinsic advantage over block codes in handling real communication line errors. The codes can be designed to accomodate bursts by adding delay between stages of the generating shift register, or they can be interleaved in the same manner as block codes, using the constraint length as the block length. Practical good convolutional codes tend to be low rate. Threshold decoding, which involves calcuation of a syndrome and is most similar to block decoding, is the decoding method of choice in a burst error environment. The popular Viterbi decoding methods (hardware is available) are recommended for use only in random error environments unless interleaved.

3.3 Hybrid Error Control Systems

A hybrid error control system adds FEC to an ARQ system to correct some error patterns, thereby reducing the frequency of block re-transmission.

Intuitively, in a hybrid system, the ARQ function should be used to repair bursts of errors, which will often exceed the FEC correction capacity, and the FEC should be used to correct the

random errors which the ARQ handles inefficiently.

The addition of FEC has several drawbacks, however. Two of these drawbacks affect throughput directly: throughput is reduced by the code rate, and the error correction processing time adds to the receiver achnowledgement response delay. There are also two hardware considerations, the cost of the coder/decoder hardware and the potential increase in continuous ARQ system data memory requirements caused by the increased response delay.

Typical performance of a hybrid ARQ system is that shown in Figure 3.4. At very low bit error rates the ARQ system has infrequent retransmissions and a throughput approaching 1, while the hybrid system throughput approaches the code rate R. At some error rate E_C, in a properly designed system, the benefits of the hybrid system's error correction and resulting reduced block retransmission rate outweigh the code overhead, and the hybrid system's throughput exceeds that of the ARQ system.

There are several reasons why the hybrid throughput might never exceed the ARQ throughput:

a) The code rate may be so low that it always outweighs the benefits of error correction.

b) If error patterns often exceed the correction capacity of the code, the reduction in the block re-transmission rate will be minimal.

c) In a go-back-N ARQ system, the increased acknowledge response time might always outweigh the benefits of error correction.

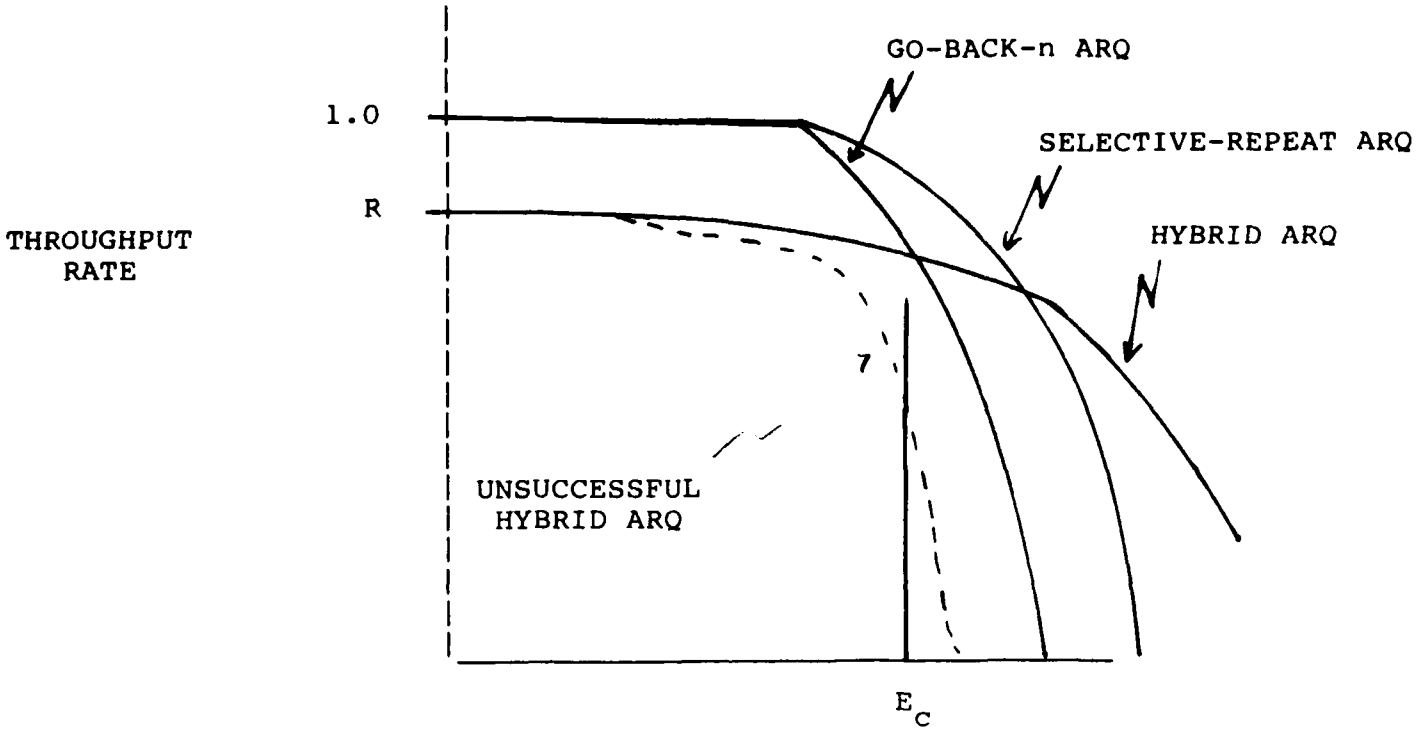


FIGURE 3.4 TYPICAL ERROR CONTROL THROUGHPUT

Assuming that the hybrid throughput curve does cross the ARQ curves, it must occur at an error rate E_C low enough that the average page throughput is improved.

3.3.1 Hybrid-ARQ Data Format

It is assumed that hybrid-ARQ system data would be transmitted in HDLC blocks for the reasons listed in Section 2.3. The HDLC information field will contain a one or two byte address field followed by coded data. If block codes are chosen, the coded data field would be an integral number of code words in length; if convolutional codes are chosen, the field will be an integral number of constraint lengths.

3.3.2 Data Coding

The hybrid-ARQ system must perform both error correction and error detection. There are many ways of implementing this. A single or compound code can simultaneously perform both functions, or they can be performed by separate codes. The codes may be block, convolutional, or a combination of these. Interleaving may be added to any configuration. The error correction capacity affects the throughput efficiency of the system, while system reliability depends upon the error detection capability.

The analysis of continuous transmission ARQ systems in Section 3.2 indicated that throughputs of about 0.8 are achievable at block error rates of 10^{-1}, and therefore the addition of FEC with code rates lower than this is unlikely to improve throughput performance.

Generally, the use of a low rate code such as the (24,12) Golay

block code or any 1/2 rate convolutional code reduces the fax data transmission rate by an unacceptable amount - it doubles the effective data transmission time, increasing the 9600 bps transmission time of CCITT document #1, Huffman coded, from 30 seconds to 1 minute. This same throughput can be achieved by reducing the uncoded transmission rate to 4800 bps. According to typical bit error rate performance data published for the Rockwell V96P/1 modem, reducing the transmission rate from 9600 bps V29 to 7200 bps V29 or 4800 bps V27 results in a 3 db improvement in signal to noise ratio. The 1/2 rate FEC is unlikely to yield more than that - data for one commercially available 1/2 rate convolutional coder/Viterb decoder shows a 3 db gain - so the use of FEC seems unjustified at these code rates.

Several data coding schemes are possible:

1. Single code

A single code can be used for both error correction (c) and detection (d). If the code has minimum distance d_{min}, then

$$c + d \leq d_{min} - 1$$

$$d_{min} \leq (n-k) + 1$$

The problem with this approach is that the number of check bits (n-k) must be large to accomodate both correction and detection. If a high code rate is also a requirement, the code block length and associated delays and hardware complexity become so large as to be

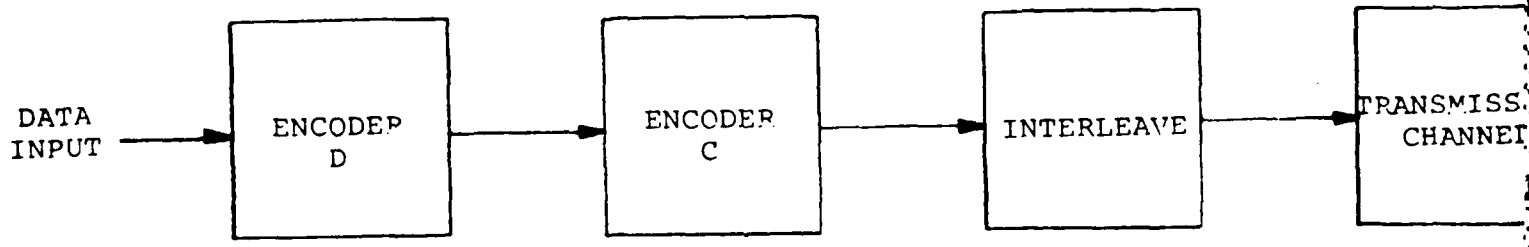
impractical. For example, it is desirable to have error detection as good as that provided by the 16-bit HDLC CRC check code. This would require a code block length of several hundred bits.

Another consideration involves the choice of the HDLC format for data blocks. Since the widely available single chip HDLC controllers all include the CRC generation/checking functions, it seems logical to continue to use that for error detection, and to add a separate code for error correction.

2. Separate error correction and detection codes.

Separate correction and detection codes might be configured as shown in Figure 3.5. The data is first coded for error detection (Code D) and the output of that encoder is coded for error correction (Code C). At the receiver, decoder C corrects those transmission errors within it's capacity, and decoder D checks for the presence of any remaining errors. Those errors that C cannot correct will be detected by D and a block re-transmission will be requested. In this sort of scheme, C is considered the 'inner' code and D the 'outer'.

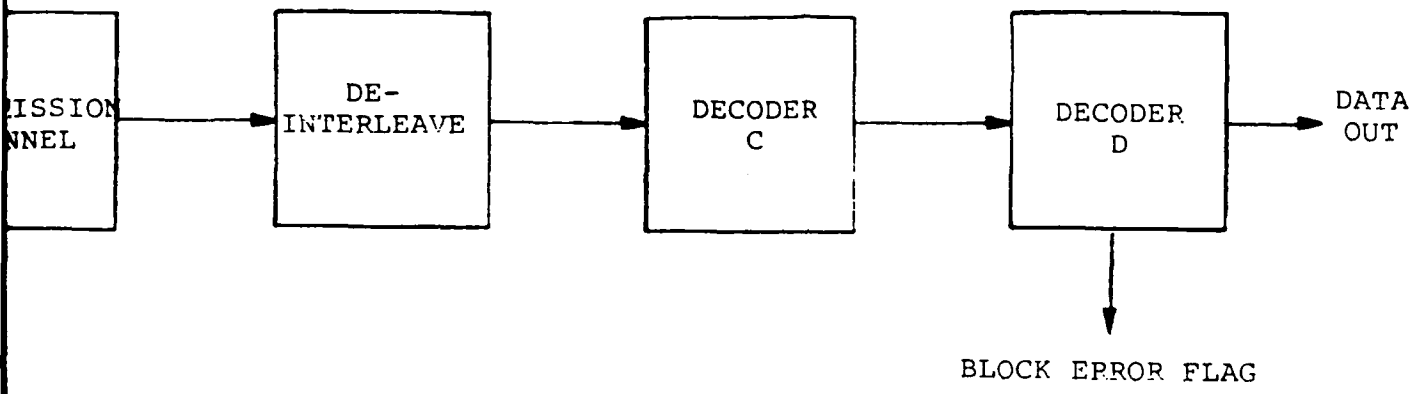
If interleaving is used to improve burst error performance, it should be inserted inside of C, as shown in Figure 3.6, because the main benefits of interleaving are in the error correction process rather



C = ERROR CORRECTION CODE

D = ERROR DETECTION CODE

FIGURE 3.5 GENERAL HYBRID E



D ERROR CONTROL DATA CODING

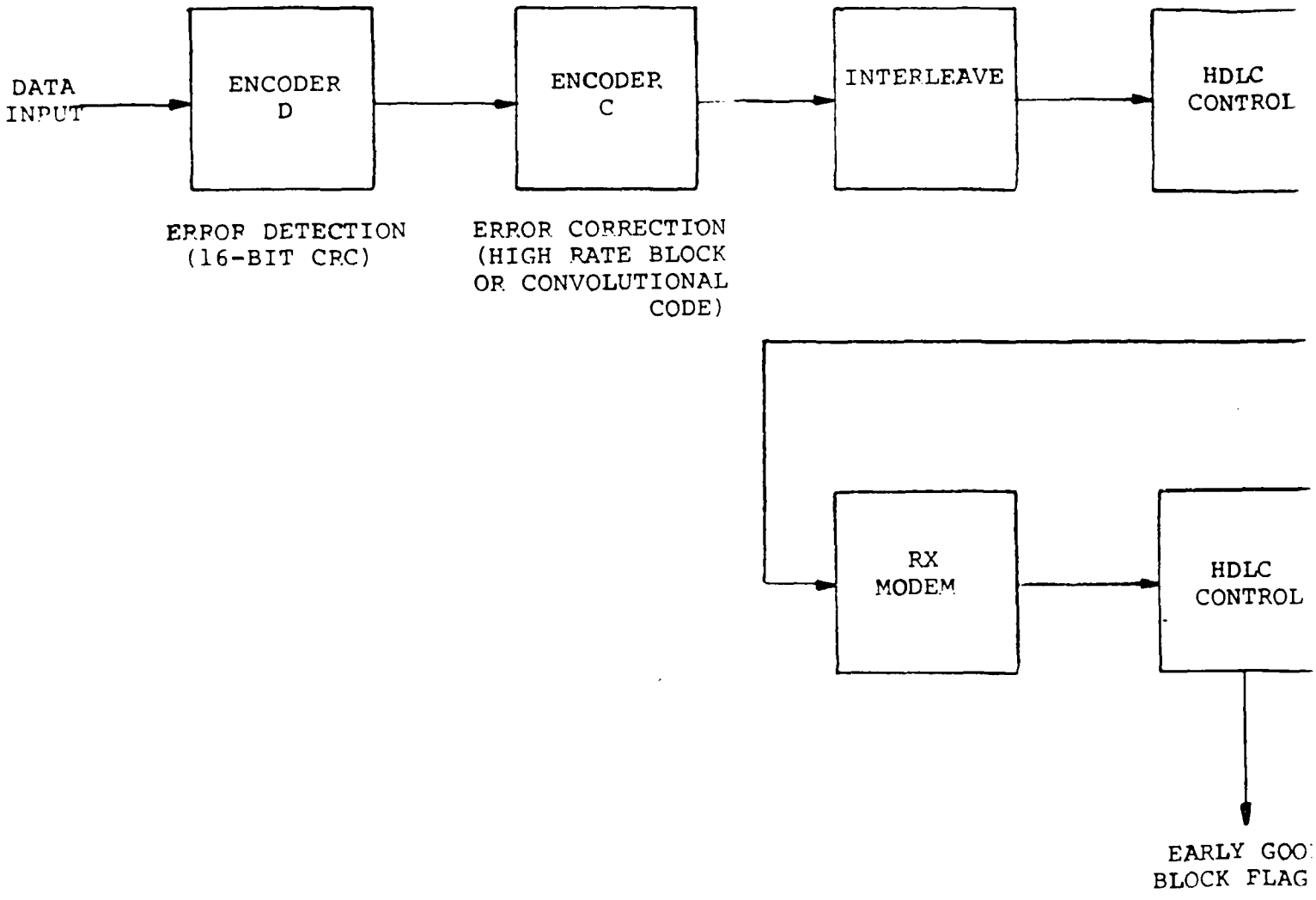
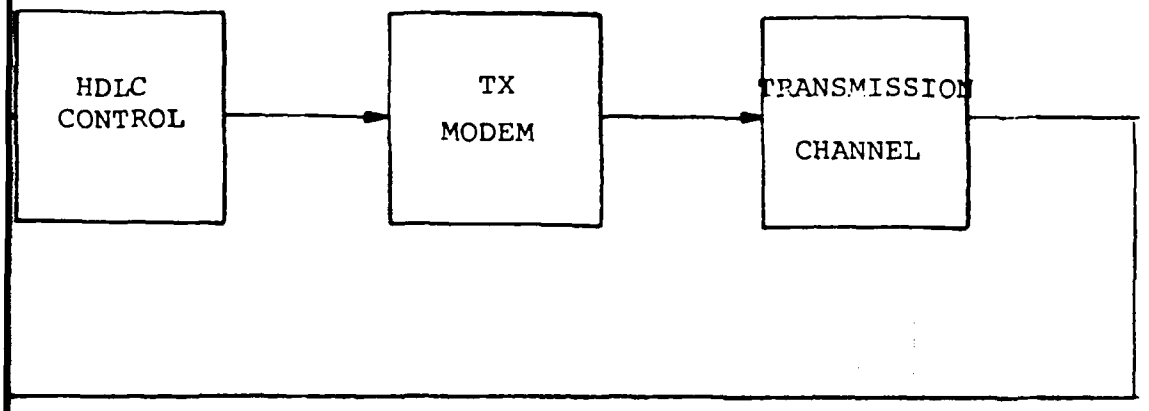
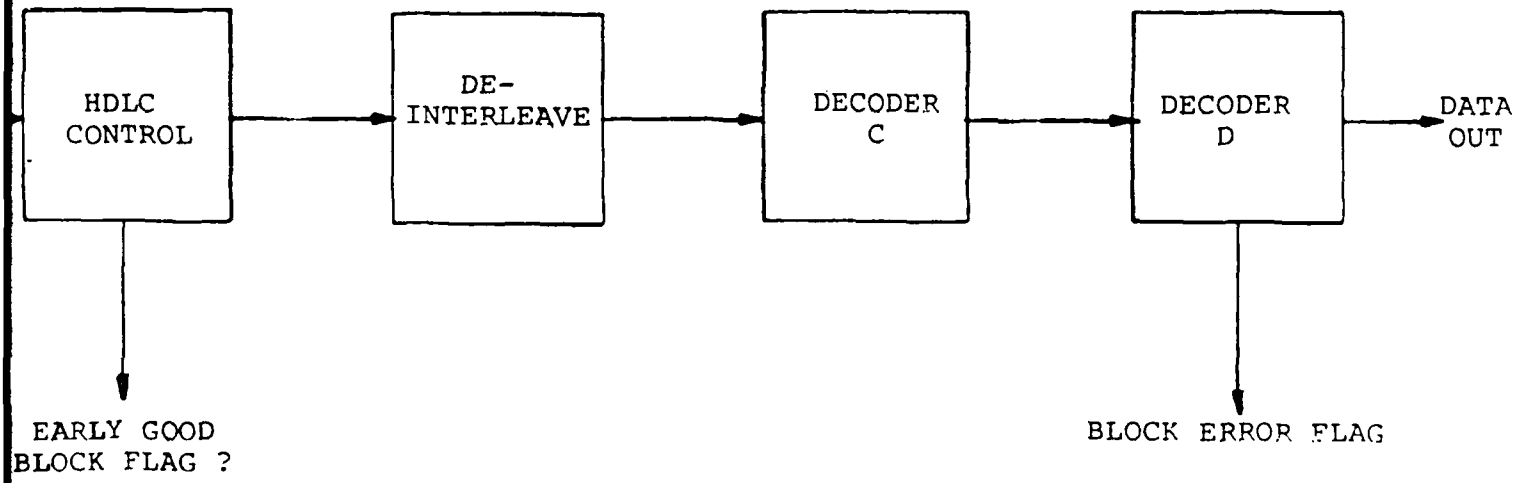


FIGURE 3.6 PROPOSED HYBRID EP





) HYBRID ERROR CONTROL DATA CODING

than in error detection.

It is tempting to consider the encoder D and decoder D blocks to be the HDLC protocol control chips, with the associated CRC check field performing the error detection function. This puts the error correction and interleaving outside the HDLC synchronization and data framing, however, and would require some other method of data synchronization.

A more practical configuration is shown in Figure 3.6.

a) Encoder/decoder D is a high rate block code designed for error detection only. One good choice for this, as we know, is the HDLC CRC check code. This is a shortened cyclic block code, with code words length equal to the data block size plus 16 check bits. There are single chips available which generate and check these codes, offering a variety of polynomials, which include that used in the HDLC protocol. Performance of this code would be the same as that detailed in Section 2.3, and the code rate with a 4K bit data block is essentially unity.

b) Encoder/Decoder C should be a high rate block or convolutional code. An analysis of throughput gain vs. code complexity is

given below.

c) The HDLC protocol is the innermost function, providing data synchronization. The CRC error check function of the HDLC protocol can be used to obtain an early indication of an error free block, and so might marginally improve ARQ response time in some designs.

3.3.3 Hybrid System Performance

It is difficult to calculate exactly the contribution of FEC in a hybrid ARQ system for the same reasons that it is difficult to calculate the performance of FEC alone: the widely varying and generally ill-defined telecommunication error distribution. Some general conclusions can be drawn, however.

Curves of throughput performance of go-back-N, selective-repeat, and page selective-repeat hybrid ARQ systems are shown in Figure 4.1 superimposed on curves of the corresponding ARQ systems. Two typical codes are used: the Hamming (15,11) and a BCD (63,51). The curves are plotted for both random errors and for the more realistic case where the FEC corrects only 90% of the block errors. It can be seen that even in the ideal random independent error environment, the addition of FEC improves throughput only as the bit error rate approaches 10^{-4}, and it is not obvious that the improved throughput under noisy line conditions justifies the increased hardware complexity and reduced throughput on good connections.

4.0 SUMMARY

The relative performances of the techniques described in this paper are summarized in Table 4.1 and in the throughput curves in Figures 4.1 and 4.2. A comparison of the normalized throughput of each technique at three different block error rates is presented in Table 4.1; this table also contains a summary of the major hardware considerations associated with each technique.

The normalized throughput for the various error correction methods is plotted as a function of error rate in Figures 4.1 and 4.2. Figure 4.1 assumes random errors and 4K bit data blocks where applicable. Note that the assumption of random errors has little impact on the analysis of the ARQ error control methods, which are relatively independent of error distribution, but is an oversimplification in the analysis of the hybrid error control methods. The effect of burst errors on the FEC component of these hybrid techniques is not taken into account when a random error environment is assumed.

Figure 4.2 shows throughput performance at 4800 bps in the noise environment documented by data from the AT&T 1969-1970 line survey. Again, assumptions were made concerning FEC performance.

TABLE 4.1

RELATIVE PERFORMANCE OF TECHNIQUES

ERROR CONTROL TECHNIQUE	GROUP 3 PROTOCOL COMPATIBILITY (1)	HARDWARE CONSIDERATIONS				APPROXIMATE NORMALIZED THRUPUT BLOCK ERROR RATE (2)			COMMENTS
		FULL-DUPLEX MODEM (2)	TX MEMORY	RX MEMORY	CODE	Ø	10^{-2}	10^{-1}	
Page Repeat	1 C	No	1 Pg or rescan cap'blty	No	No	1	Ø.85	Ø.55	For 1% line error retransmit criteria. Not really comparable with others since no errors were corrected.
Stop-&-Wait ARQ	2 B	No	1 block	1 block	No	Ø.29	Ø.29	Ø.26	Very slow. Large response overhead even at low error rates.
Go-Back-N ARQ	3 A,B	Yes	N blocks to 1 pg.	2 blocks	No	1	Ø.96	Ø.69	
Selective-Repeat ARQ	3 A, B	Yes	N blocks to 1 pg.	N blocks to 1 pg.	No	1	Ø.99	Ø.9Ø	Best throughput
Page Selective Repeat ARQ (4)	2	No	1 pg	1 pg	No	1	Ø.94	Ø.78	Good compromise. Nearly as fast as continuous transmission ARQ without requiring modem back channel.

ERROR CONTROL TECHNIQUE	GROUP 3 PROTOCOL COMPATIBILITY (1)	HARDWARE CONSIDERATIONS				APPROXIMATE NORMALIZED THRUPUT BLOCK ERROR RATE (2)			COMMENTS
		FULL-DUPLEX MODEM (2)	TX MEMORY	RX MEMORY	CODE	Ø	$1Ø^{-2}$	$1Ø^{-1}$	
FEC (39,32) rate Hamming Code	2	No	No	No	Yes (Single Chip Implementation)	Ø.82	Ø.82	Ø.82	Single error correction. Corrects almost all random errors, but falls apart in presence of bursts. Single chip hardware widely available. Interleaving helps burst performance.
FEC - 1/2 rate convolutional	2	No	No	No	Yes Complex Implementation	Ø.5	Ø.5	Ø.5	Uncertain what percentage of errors this corrects or whether it is better than 2 to 1 reduction in transmit bit rate.
HYBRID -- Go-back-N ARQ with FEC (3)	4	Yes	N blocks to 1 pg.	2 blocks	Yes	Ø.87	Ø.87	Ø.75	Exact analysis difficult. Does not seem to be enough potential for improvement to warrant added complexity and reduced throughput on good connections.
HYBRID -- Selective Repeat ARQ with FEC (3)	4	Yes	N blocks to 1 pg.	N blocks to 1 pg.	Yes	Ø.87	Ø.87	Ø.85	
HYBRID -- Pg. Selective Repeat ARQ with FEC (3)	3	No	1 page	1 page	Yes	Ø.87	Ø.85	Ø.79	

(1) 1 (least change) to 4 (most change)
A - HDLC data block format
B - 300 bps ACK/NACK responses on block level
C - 300 bps ACK/NACK on page level (less complex than B)

(2) Does not include HDLC overhead of approximately 4% Assumes 4K bit blocks.

(3) Based optimistically on a 7/8 rate code that corrects 75 % of block errors.

(4) Assumes 3 second interpage delay.

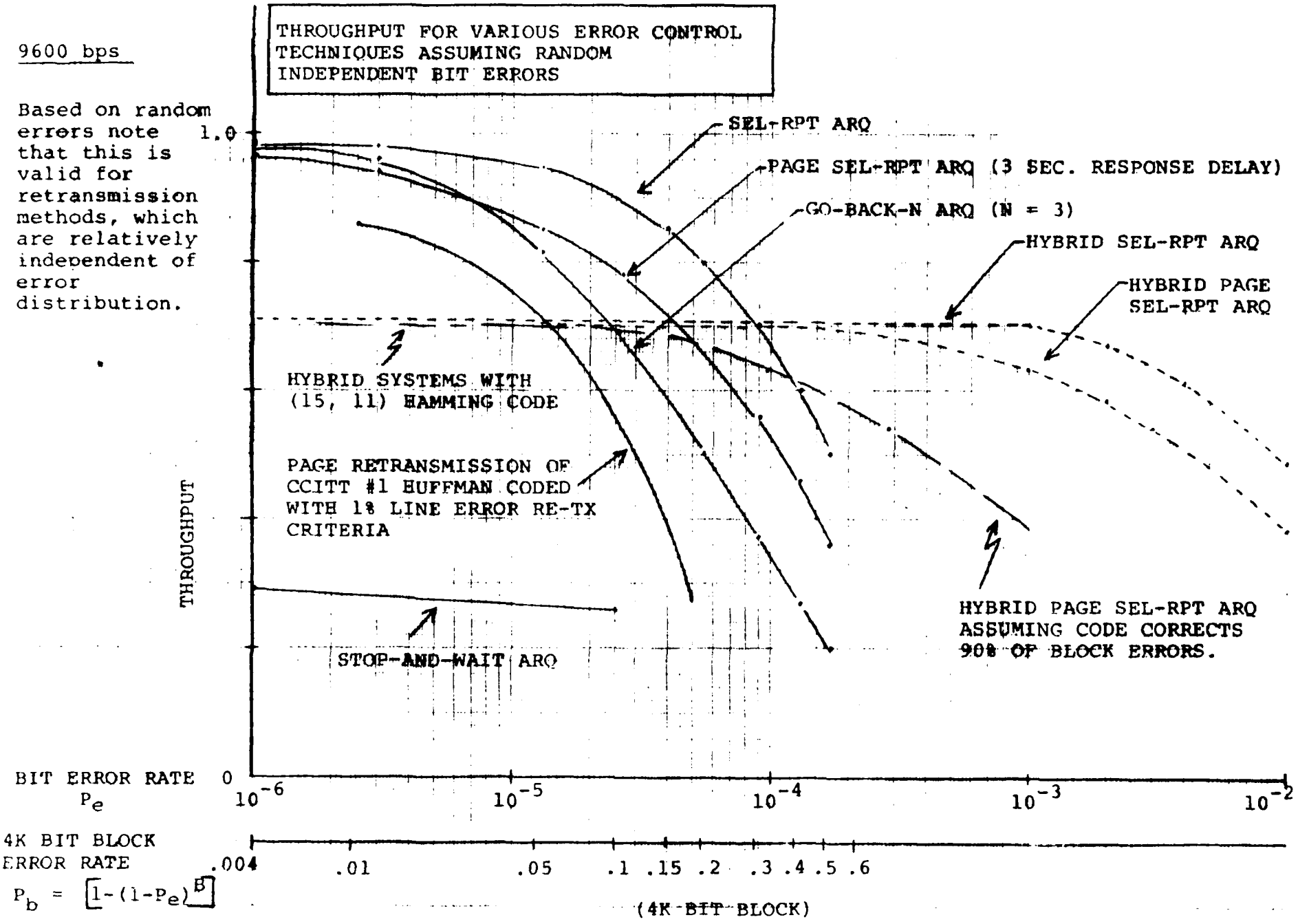

THROUGHPUT FOR VARIOUS ERROR CONTROL TECHNIQUES ASSUMING RANDOM INDEPENDENT BIT ERRORS
9600 bps
Based on random errors note that this is valid for retransmission methods, which are relatively independent of error distribution.
SEL-RPT ARQ
PAGE SEL-RPT ARQ (3 SEC. RESPONSE DELAY)
GO-BACK-N ARQ (N = 3)
HYBRID SEL-RPT ARQ
HYBRID PAGE SEL-RPT ARQ
HYBRID SYSTEMS WITH (15, 11) HAMMING CODE
PAGE RETRANSMISSION OF CCITT #1 HUFFMAN CODED WITH 1% LINE ERROR RE-TX CRITERIA
HYBRID PAGE SEL-RPT ARQ ASSUMING CODE CORRECTS 90% OF BLOCK ERRORS.
STOP-AND-WAIT ARQ
THROUGHPUT
1.0
0
BIT ERROR RATE
P_e
10^{-6}
10^{-5}
10^{-4}
10^{-3}
10^{-2}
4K BIT BLOCK ERROR RATE
$P_b = [1-(1-P_e)^B]$
.004
.01
.05
.1
.15
.2
.3
.4
.5
.6
(4K BIT BLOCK)

FIGURE 4.1

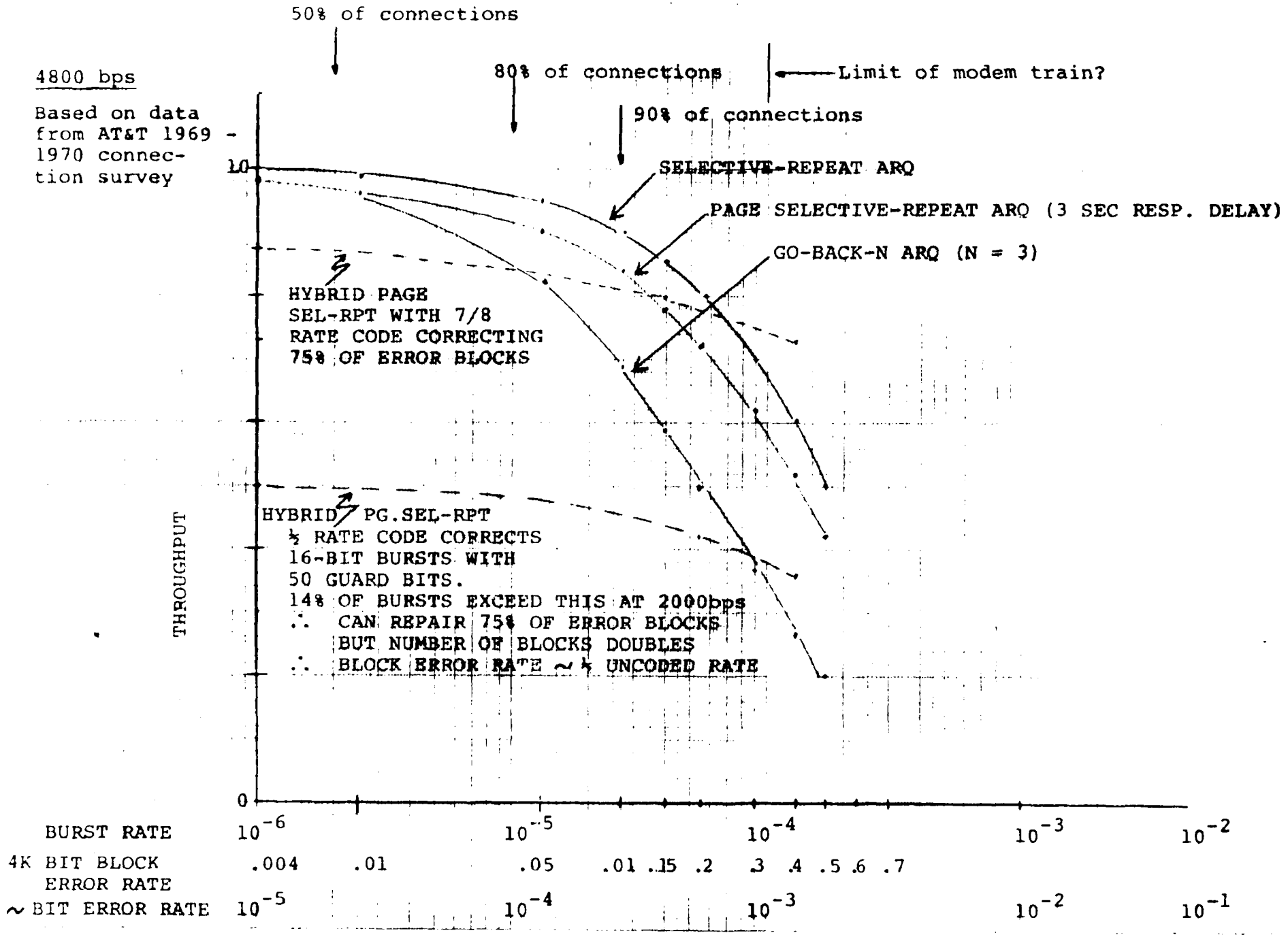

50% of connections
80% of connections
90% of connections
Limit of modem train?
4800 bps
Based on data from AT&T 1969 - 1970 connection survey
SELECTIVE-REPEAT ARQ
PAGE SELECTIVE-REPEAT ARQ (3 SEC RESP. DELAY)
GO-BACK-N ARQ (N = 3)
HYBRID PAGE SEL-RPT WITH 7/8 RATE CODE CORRECTING 75% OF ERROR BLOCKS
HYBRID PG.SEL-RPT
½ RATE CODE CORRECTS 16-BIT BURSTS WITH 50 GUARD BITS.
14% OF BURSTS EXCEED THIS AT 2000bps
∴ CAN REPAIR 75% OF ERROR BLOCKS
BUT NUMBER OF BLOCKS DOUBLES
∴ BLOCK ERROR RATE ~ ½ UNCODED RATE
THROUGHPUT
1.0
0
BURST RATE
4K BIT BLOCK ERROR RATE
~ BIT ERROR RATE
10^-6
10^-5
10^-4
10^-3
10^-2
.004
.01
.05
.01 .15 .2 .3 .4 .5 .6 .7
10^-5
10^-4
10^-3
10^-2
10^-1

FIGURE 4.2

REFERENCES

1. "Errors and Error Control", Burton, H. O. and Sullivan, D. D., Proceedings of the IEEE, November, 1972, pp. 1293-1301.

2. "Optimal Message Block Size for Computer Communications with Error Detection and Retransmission Strategies", Chu, W. W., IEEE Transactions on Communications, October, 1974, pp. 1516-1525.

END

9-87

DTIC